Some Issues in the Testing of Computer Simulation Models

David J. Murray-Smith,
School of Engineering, Rankine Building, University of Glasgow,
Glasgow G12 8QQ, Scotland, UK.
david.murray-smith@glasgow.ac.uk

Abstract. The testing of simulation models has much in common with testing processes in other types of application involving software development. However, there are also important differences associated with the fact that simulation model testing involves two distinct aspects, which are known as verification and validation. Model validation is concerned with investigation of modelling errors and model limitations while verification involves checking that the simulation program is an accurate representation of the mathematical and logical structure of the underlying model. Success in model validation depends upon the availability of detailed information about all aspects of the system being modelled. It also may depend on the availability of high quality data from the system which can be used to compare its behaviour with that of the corresponding simulation model. Transparency, high standards of documentation and good management of simulation models and data sets are basic requirements in simulation model testing. Unlike most other areas of software testing, model validation often has subjective elements, with potentially important contributions from face-validation procedures in which experts give a subjective assessment of the fidelity of the model. Verification and validation processes are not simply applied once but must be used repeatedly throughout the model development process, with regressive testing principles being applied. Decisions about when a model is acceptable for the intended application inevitably involve some form of risk assessment. A case study concerned with the development and application of a simulation model of a hydro-turbine and electrical generator system is used to illustrate some of the issues arising in a typical control engineering application. Results from the case study suggest that it is important to bring together objective aspects of simulation model testing and the more subjective face-validation aspects in a coherent fashion. Suggestions are also made about the need for changes in approach in the teaching of simulation techniques to engineering students to give more emphasis to issues of model quality, testing and validation.

Keywords: validation, verification, transparency, documentation, control engineering, education

1. Introduction

In using a model as a basis for making predictions, or for design purposes, we must have confidence that results obtained from the model will be useful. This means that properly defining the requirements is an essential first step, just as in all other areas of software development. However, the simulation model always differs from reality and it is always important for the user to understand the limitations of a model and to know how accurate its predictions will be in different circumstances. Testing may show that errors are present but can never prove that they are absent. Some form of risk assessment and decision making is therefore always necessary in order to establish how much or how little testing is appropriate for a simulation model and this must always depend on how the model is to be used and the specific requirements of the final application.

I have been involved with simulation for all of my professional career and during the1990s I became very concerned that insufficient attention was being given to issues of model quality, especially in the education of engineering students, and started to try to do something more positive about this in the context of the courses that I was teaching. Those courses, at that time, were mainly on modelling and simulation for engineering applications and in control engineering, where much use is made of mathematical models and simulation methods.

My concern about issues of model quality and testing came about partly from observing differences in approach to simulation program development and testing shown by students who had taken a software engineering course as part of their degree programme, as compared with the majority who did not have the benefit that computing science background. The students who had more software development experience within computing science degree programmes undoubtedly had a more systematic approach to testing and saw documentation of tests as being important. In contrast, most of the other students tended to be over-confident about their models, often on the basis of a few subjective assessments of the overall behaviour for a few (often fairly randomly chosen) sets of test conditions. They also tended to have a haphazard approach to documentation of testing and tended to regard it as an extra (and somewhat unnecessary task), rather than being an important part of the model development process. It should be noted that all the students, whatever their backgrounds, had been given the same guidance in tackling practical modelling and simulation assignments.

These simulation modelling courses involved physically-based dynamic models, generally with nonlinear ordinary or partial differential equations or differential algebraic equations. The underlying models could also give rise to linearised descriptions for specific sets of operating conditions and to experimentally-derived models obtained using system identification and parameter estimation methods. In some cases, real-time models were required which could be used for hardware-in-the-loop testing. Thus, a given system could give rise to a number of different models, all of which were closely connected but all of which had a specific use.

Getting the across the idea to students that one could have a number of different forms of model for one given system was not a difficulty, but getting the students to appreciate the limitations and applicability of each of the different models and the way in which the different types of model could be used in engineering analysis, design and simulation was more challenging. Also, issues of transparency of the processes involved in model development and testing and also questions of model management immediately became evident. Especially significant was the difficulty of ensuring that model documentation was kept up-to-date and that accepted changes in the underlying physically-based nonlinear model were carried over to the other (e.g. linearised or real-time) versions of the model. Indeed, in group projects involving a number of students working together, the importance of version control for models very rapidly became very evident and brought home to the students the fact that proper management of models and data sets was essential.

In 2001 I presented a paper entitled “The Validation of Computer-Based Models in Engineering: Some Lessons from Computer Science” at a conference in Glasgow and a modified version subsequently formed the basis of a journal publication [1]. In that paper I suggested that questions of overall quality in computer-based models and formal processes of model testing were seldom given more that passing attention in engineering reports and technical publications. I pointed out that this unsatisfactory situation contrasted sharply with approaches to issues such as version control, testing and documentation in more conventional software projects. That discussion about problems of simulation model quality and testing was consistent with statements about simulation model development made by others (e.g. [2-4]) and the whole topic of testing and validation of simulation models has been the subject of more detailed discussion recently (e.g. [5] [6]).

2. Specific Issues in Testing Simulation Models

Although there has been a gradual increase in the general awareness of model developers and the user population in the importance of model validation and testing over the years, the extent to which ideas are put into practice varies greatly from one organisation to another. The situation regarding testing of models still seems to me to be unsatisfactory in terms of teaching in many university courses. Too often, in engineering, the attitude in the past has that “this is how we always have done it in terms of checking our models” or “this model is based on sound physical principles so it must be right” or “we know there may be some problems but we will sort them out as we go along”. Too often, also in engineering, we read of important design contracts where major changes have been found to be necessary at a late stage in the development of the project, with a resultant major increase of cost and significant delays in terms of the project delivery. Clearly, in such cases, things have not worked out as expected and in many cases this has been associated with inadequacies in terms of modelling and simulation at early stages of the project.

Like many other areas of software development, computer-based model development requires an intimate knowledge of the intended application, a systematic approach and considerable ingenuity, expertise and insight throughout the development and in the design of tests. The qualities required by computing professions in software development and testing and by engineers engaged in the development, testing and application of computer-based models are certainly similar in many ways. For example, as in the development of other software, the process of testing in simulation model development is more than just a phase of work that occurs at the end of the development cycle. The testing has to start during the definition of the model requirements and must also be a feature of every stage of the development process.

As in software testing generally, model testing usually involves a "*bottom up*" type of procedure in which models of individual components of the system are tested first. Integration testing and complete system testing are carried out at a later stage of model development. *Regressive testing* procedures in which earlier tests are re-executed when changes are made in the model are as important in simulation as in any other field of software development.

However, we also need to examine more carefully some specific issues that can arise in simulation model testing that do not really have a counterpart in the more general software testing process. The first point to note is that the simulation model testing process really has to be split into two parts. The first of these (often termed "verification") involves checking that the simulation program matches the underlying mathematical or conceptual description of the system being considered. The second part of the process for testing of simulation models ("validation") involves comparison of the model and the real system which it represents to ensure that the simulation model behaviour adequately matches the behaviour of the real system in terms of aspects that are important for the intended application.

2.1 Verification of Simulation Models

The *verification* process (sometimes referred to as "*internal verification*" to emphasise that this aspect of testing is internal to the simulation model) is essentially a process of traditional software testing and debugging to ensure that the simulation program is free from logical and coding errors and that appropriate numerical algorithms have been applied correctly to the problem in hand. This is very close to the testing process within many other complex software development projects and standard techniques of software testing (see, e.g., [7], [8]) can be applied. It is a relatively straightforward process and it has been suggested that formal methods (and especially "lightweight" formal methods) could be useful in dealing with some issues.

2.2 Validation of Simulation Models

Validation is a much more difficult and open-ended problem than those encountered in the internal verification processes. It may be referred to as "*external validation*" to emphasise that this aspect of testing is based on information from the real world. The amount of validation testing necessary for a given model depends on the consequences of possible model errors and is therefore a matter for risk assessment of some kind in the context of the intended application. In safety-critical application areas model testing and documentation is already much more rigorous than in other types of application.

A model is only an abstraction of some features of a real system which the model is intended to represent. The modelling process inevitably raises important philosophical questions but the most important issue, whatever the intended application, is to determine the level of model fidelity needed for that application. Models must also be transparent so that those using them have a clear understanding of how the model was developed, how it is organised and what its limits are. Users thus need to understand the "neighbourhood of validity" of the model rather than any unique or precisely-defined set of conditions for its use.

The tasks involved in the development of a simulation model generally extend far beyond the technical processes of constructing a computer-based description of a set of mathematical equations and logical statements. Investigation of the accuracy and limitations of a model may include analysis of linearised descriptions derived from a more general nonlinear model, together with storage, retrieval and quantitative comparison of simulation and experimental

results for a wide range of conditions. It may also involve system identification and parameter estimation to establish the suitability, or otherwise, of structural assumptions or parametric values used in a physically-based model, sensitivity analysis, experimental design, post-processing of experimental data and visualisation. Carrying out these tasks may involve generation of a considerable amount of data and all of this must be kept track of and documented. External validation of simulation models also often involves some elements of *face validation* where the opinions of experts are used to establish the extent to which a specific model is credible and to establish aspects of the model that need to be improved.

In many engineering projects modelling is used for design purposes and simulation allows "what if" situations to be postulated and possible trade-off studies to be carried out looking at different design options. At the early stages in the life cycle of an engineering project there is little prospect of being able to validate models in a detailed quantitative sense and error bounds on predictions are large. However, previous experience with other projects using broadly similar simulation models may provide useful insight and may offer ways forward. As the new project proceeds simulation models become more fully integrated into the design process and the accuracy of the models being used should increase, along with the confidence of the members of the design team. At some point in the project the direction of information flow, which is initially from the simulation models to the design, changes as data become available from elements of the real system and testing of sub-systems and experimental prototypes begins. The bi-directional transfer of data then continues at all later stages of development, with models being updated regularly to help to ensure that the product meets the specification. The whole test process must, in general, be repeated many times.

When one starts to look carefully at the issues that arise in the validation of simulation models in different fields it becomes clear that, although the problems of validation may appear very different, many of the same issues that crop up with one application can also arise in the validation of models of other complex engineering systems. There are similar problems of measurement since key variables of the model may not correspond to readily accessible variables of the real system. There can also be significant problems of structural and parametric uncertainties. Measurement noise and the associated need for pre-processing of experimental data is a feature of most applications, as are problems associated with the drawing of boundaries around the system and the splitting of a complete system model into appropriate sub-systems. One of the most significant issues in this field is how best to integrate the results of face validation and the subjective (and often differing) opinions from a number of subject experts with whatever results are available involving quantitative comparisons of system and model responses. Also, questions of model credibility introduce more problems the more generic the model is intended to be since a generic model must allow for many different applications.

In order to look at some of these issues in a little more detail let us now consider an application which is typical of some of those that arise in engineering. This does not involve a complete system design but is concerned with the modelling of an existing mechanical system in order to make major changes to its control system. Such cases are commonly encountered when digital-processor based control is added as part of a mid-life system upgrade.

2.3 An example: development and testing of a simulation model of a hydro-turbine generator system

This case-study, which I have used in teaching for Master's degree level courses, is based on experience gained within the University of Glasgow during a research project carried out with partners from the electrical power industry [5]. However, I cannot claim that, at the time of the original research, the procedures adopted for testing of our simulation models were particularly good. Indeed, the point of developing this case study was to highlight difficulties which, with the benefit of hind-sight, might have been avoided or reduced had a more systematic, rigorous and transparent testing and model management processes been adopted.

The context of the original work was the development of a simulation model of an existing power station involving a reservoir, pipeline, hydro-turbine, generator system and associated electrical power distribution network. The power station had been designed and built during the decade immediately following the end of the Second World War and was originally intended to supply part of the base electrical load in Scotland. This meant that the associated control systems were designed for (more or less) steady-state operation and changes in

operating conditions were expected to be relatively slow. In more recent years, requirements have changed and hydro-electric power generation in Scotland is seen as being more importance in terms of meeting short-term changes in demand rather than supplying part of the base load. As a result, there has been interest within the electrical power industry in examining the potential for modifying existing hydro-power stations to provide a rapidly-responding source of energy to meet large changes, caused for example by failure of a generator within a conventional fossil-fuel or nuclear station. Interest in faster-acting hydro-turbine speed control systems started to grow and attention was directed to electronic control systems as a possible flexible replacement for the existing mechanical governor systems that had been optimised for conditions involving a steady-state operating point close to full load.

These replacement speed control systems could involve analogue or digital governor hardware and were to be interfaced to existing mechanical linkages connected to the turbine guide vanes through electro-hydraulic actuators. In terms of the dynamics of the existing systems, little was known in detail about the frequency response of the system being controlled and one of the major areas of concern in investigating possible new designs of faster-acting governors was the possibility of pipeline damage due to resonance effects. Before any testing of new governors could be considered it was essential that the governors should be fully-investigated off-site using a real-time simulation of the system, but such a simulation model did not exist and had to be developed from information available about the existing power station and from a limited site testing programme agreed in detail by engineers from the electrical supply authority that operated the station.

The main difficulty in the development of the simulation model was the extent of uncertainty about the real system, especially in terms of its dynamic characteristics. Figure 1 is a schematic diagram of the complete system showing some elements included in the model. The operating conditions for the system involved different power levels and included situations where the generator supplied a local isolated load as well as the more normal case where the generator was coupled to the national distribution network.

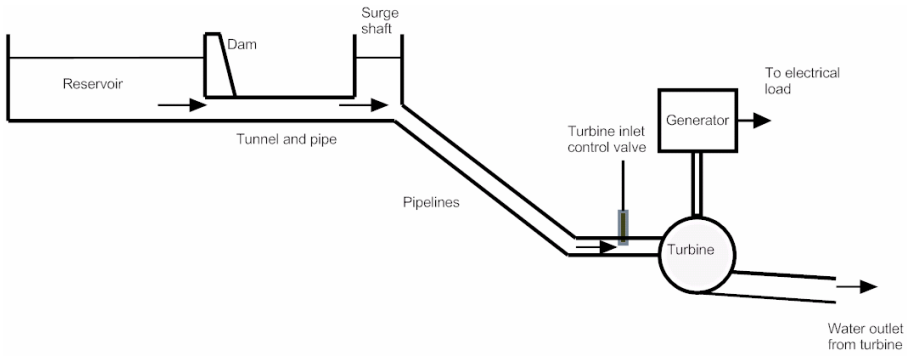

Figure 1: Schematic diagram showing pipeline, turbine and generator system [5].

The most demanding sub-system in terms of the real-time simulation was the pipeline network. Although it appears as a single tunnel and pipe in Figure 1 the system had a more complex structure due to the fact that it involved a branching system of pipes leading to four turbines. The complete pipeline system had been modelled previously using a description based on partial differential equations but this could not be incorporated into the real-time simulation using the available computing facilities. In order to achieve satisfactory performance in real time the pipeline system had to be approximated by a lumped parameter model.

Variables available for measurement in the real system included turbine speed, power level and a number of mechanical variables. Opportunities for carrying out tests on the real system were, however, rather limited. Operational and safety constraints, associated particularly with the integrity of the pipeline system, meant that testing had to be carried out over a very limited frequency range to avoid possible resonances and associated water-hammer effects. Nevertheless, some limited frequency response testing was possible and testing was also carried out in the time-domain using test signals that did not involve significant content in the range of frequencies that was believed to present a risk for the pipeline. These initial tests provided useful information which allowed some important parameters to be estimated for which reliable values had previously been missing and also provided a basis for checks of

model structure and nonlinearities. The new information gathered was incorporated into the model which had a physically-based mathematical and logical structure.

Following the initial tests carried out for system identification and parameter estimation purposes, the simulation model was tested in conjunction with a representation of the existing relatively slow type of mechanical governor. Comparison of measured and simulated results for the closed-loop situation showed general agreement and this allowed some confidence to be established in the simulation model. It is important to note that the data sets used for these validation tests were different from the data sets obtained from the tests previously carried out for estimation of model parameters.

Although quantitative comparisons of the model and real system were very encouraging, it was suggested that there should be further testing of the real-time simulation by some of the operators who had many years of experience at the power station. This hands-on face validation exercise highlighted some important deficiencies in the model for specific operating conditions. These issues had not been detected from the other, more quantitative and supposedly more objective, tests. The problems were associated with the effects of nonlinearities in the linkages to the turbine inlet. Inaccurate representation of backlash effects in the model meant that, under some operating conditions, limit cycle phenomena would occur that were much more pronounced in the real system than in the model. It became clear, subsequently, that these limit cycles were present in other test data but were not obvious because the duration of the test records was too short in many cases. Suggestions made by the operating staff led to further testing at the power station, more detailed investigation of the relevant nonlinear effects and to modifications within the simulation model. Following the changes in the model, further face-validation tests suggested that the real-time simulation model was acceptable and could be used as a test-bed for the evaluation of the faster-acting governor hardware.

One area in which quite a lot of effort was required was concerned with the representation of the relief valve. This is an important safety feature as the valve is designed to open under conditions in which the turbine inlet closes quickly and water can no longer flow through the turbine. Without the relief valve excessive pressure could build up within the pipeline so accurate representation of this feature of the system was vitally important if the simulation model was to be used in evaluation of the new governor systems.

It is very important to note that further internal verification tests should always be carried out whenever changes are made in the simulation model. However, these internal verification tests need not involve the complete simulation model if it has been structured appropriately using well-defined sub-models.

Experience with the faster-acting analogue electronic and digital speed control systems using the real-time simulation allowed potential problems to be investigated prior to any testing of the new governor systems on site. In particular, the real-time simulation allowed possible difficulties with the pipeline system to be looked at carefully for each type of control system configuration as well as the investigation of the potential benefits of these systems in terms of the speed of response to changes of power demand and grid frequency. Once they had been evaluated using the simulation model, several types of fast-acting governor were approved for further testing on site and were installed at the power station for prolonged periods of testing, including normal service conditions in some cases.

Although the brief account given above gives relatively little information about the simulation model itself, it can be seen that the testing of the model involved a number of important aspects in addition to the processes of internal verification, thus going beyond the procedures normally encountered in software testing. These additional aspects included:

a) Initial testing on site for the purposes of model development had to be carried out using a conventional control system configuration involving the traditional slow mechanical governor. Tests of the basic simulation model for the purposes of external validation involved a representation of this existing control system, but were not based on the test data used for the estimation of parameters.

b) In this application there were constraints on tests that could be performed on the real system due to the risk of pipe-line damage. Optimisation of the design of experiments to maximise the system dynamic response over appropriate parts of the frequency range, while avoiding critical frequencies, could therefore be very important. In retrospect, experience with other applications (see, e.g., [5]) suggests that more effort applied to test signal design could have resulted in a significant increase in the efficiency and effectiveness of the site testing for the turbine system.

c) A number of different operating conditions had to be considered so a significant quantity of tests data had to be collected for parameter estimation or stored for comparison with equivalent simulation model responses. In some cases, pre-processing of data was necessary, using appropriate filtering techniques. Management of the large number of data sets created in this way is an important issue in developing a complex simulation. Careful definition of model requirements, thorough planning and sound management are required from the outset.

d) In this application face validation proved to be very important and results from face validation led to significant changes in the simulation model structure and parameters. Getting the appropriate balance between quantitative validation methods and face validation is important. In this application the results from face validation led to a better understanding of the need to match the length of measured response records to time scales considered important by those familiar with the operation of the plant.

e) Hands-on experience with the real-time simulation by the plant operators also proved important in the sense that they were able to comment on the "feel" of the simulated system in a number of different situations with which they were familiar. It is believed that simply providing recorded responses from the simulation could not have provided the same useful feedback about model deficiencies.

3. Conclusions

It is clear that the testing of simulation models involves more than just the testing of software. The process also has to involve more basic procedures to establish whether or not the underlying mathematical and logical model is an adequate representation of the real system. That process must always be carried out in the context of the intended application of the model and with careful regard to the definition of the model requirements established at the start of the project.

Conventional software testing techniques are clearly important for internal verification of simulation models. This must include line-by-line checks to ensure that the simulation program code corresponds exactly to the structure and parameters of the underlying mathematical and logical model of the system. It must also include checks of algorithms that are specific to simulation models, such as those used for integration, discontinuity detection etc. On the other hand, the external validation process is more complex and goes beyond conventional software testing concepts and requires deeper understanding of the real system being modelled. Nevertheless, many ideas that are commonly used in software testing such as regressive testing are just as relevant to external validation as they are to internal verification. Adherence to good principles of simulation model management and data management is essential to ensure that the whole model testing process, which is inherently iterative in nature, is carried out effectively. Transparency is an essential feature of simulation model testing, as is sound management of model versions and test data sets, especially when a number of people are working together as a team [6]. Every aspect of model testing must be fully documented, just as it should be in other types of software testing.

The example used in Section 2.3 provides a useful illustration of the testing of a physically-based model using direct comparisons of sets of measured data from the real system with sets of corresponding data from the simulation model, together with evidence from face validation. In this example, which is typical of engineering applications in which a model is being developed for an existing system, the simulation model was found to be inadequate from face validation testing although other comparisons had suggested a satisfactory model performance for the intended application.

Results from eternal validation never provide a unique result and can never "prove" that a model is in any sense "correct". Nevertheless, concepts from the field of software engineering and software testing can be useful within the external validation process and can be used to demonstrate, broadly, whether or not a specific model meets the given set of model requirements and can provide information about the limitations of any accepted model.

Over the past fifteen years it is undoubtedly true that more use is being made of version control in the management of simulation models and more emphasis is being given to the need for good documentation, not only of the models themselves but also of test data sets used for parameter estimation and model validation. Ideas that have been well established for some time in safety-critical application areas such as aircraft systems are now being adopted

in other fields, although the associated tools and methods are sometimes viewed as imposing significant additional burdens and extra costs on developers. Although establishing model credibility is vitally important, it is of course equally important that model management processes should be seen as helpful by their users and not become a barrier to creative thinking. Getting the right balance for each project is essential and to help to do this it is very important to fully understand what is involved and some of the potential pitfalls.

One important issue concerning the testing of simulation models relates to the education and training of those involved in model development. As mentioned in Section 1, too few students regard comprehensive testing of models and the associated model documentation as being very important. This clearly requires a change of emphasis in all courses that deal with modelling and simulation so that these aspects of the work receive more attention. Those concerned with the supervision and assessment of individual and group project work must also ensure that whenever students approach problems using simulation and modelling methods their reports include an adequate record of testing methods and results with clear justification for adoption of any specific model upon which later results depend.

References

[1] Murray-Smith, DJ. The validation of computer-based models in engineering: some lessons from computing science". *Acta Polytechnica*, 2001, 41 (4/5): 45-48.

[2] Brade, D. *A Generalized Process for the Verification and Validation of Models and Simulation Results.* Disssertation. Fakültat für Informatik, Universität der Bundeswehr, München, Germany. 2003.

[3] Pace, D.K. Modeling and simulation verification and validation challenges. *John Hopkins APL Technical Digest*, 2004, 25(2): 163-172.

[4] Ericsen, T. Physics based design, the future of modelling and simulation. *Acta Polytechnica*, 2005, 45 (4): 59-64.

[5] Murray-Smith, DJ. *Testing and Validation of Computer Simulation Models: Principles, Methods and Applications*. Cham, Switzerland; Springer, 2015.

[6] Murray-Smith, D.J. Issues of transparency, testing and validation in the development and application of simulation models, *Simulation Notes Europe*, 2016, 26(2): 57-66.

[7] Kit, E. Software *Testing in the Real World*. Harlow, UK; Addison-Wesley. 1995.

[8] Kaner, C, Falk, J and Hung Quoc Nguyen. *Testing Computer Software*, 2nd Ed.; New York, USA, 1999.